1 Logistic Function and Log Odds

The logistic function (typically also known as the sigmoid function) plays an crucial role in many classification algorithms. Formally, it is defined to be $\sigma(z) = 1/(1 + \exp(-z))$. Intuitively, this appears when we want to squash the output of our linear mapping $z = \mathbf{w}^\top \mathbf{x}$ into $[0, 1]$. At this moment, our choice of the squashing function seems arbitrary. This problem shall justify our choice more formally.

Problem 1. Let the number of classes $K = 2$ and $\mathbf{x} \in \mathbb{R}^M$ be an input with target $t \in \{1, 2\}$ (which differs from the lecture convention of $t \in \{0, 1\}$). Recall that in a discriminative model, we want to infer $p(t = 1|\mathbf{x})$. Let us start massaging this quantity by dividing it by 1

$$p(t = 1|\mathbf{x}) = \frac{p(t = 1|\mathbf{x})}{p(t = 1|\mathbf{x}) + p(t = 2|\mathbf{x})}.$$

Now observe that $\exp \ln(z) = z$. Show that $p(t = 1|\mathbf{x}) = \sigma(z)$ where

$$z = \ln \left(\frac{p(t = 1|\mathbf{x})}{p(t = 2|\mathbf{x})} \right).$$

Problem 2. The term z is known as the log odds or logits. Crucially, the act of logarithmizing the odds $p(t = 1|\mathbf{x})/p(t = 2|\mathbf{x})$ results in exactly half of $\mathbb{R}$ favoring $t = 1$ and another half favoring $t = 2$. If this is new to you, check out this video by StatQuest for a quick introduction to log odds. Intuitively, this equal favoring of both classes makes for a favorable quantity to learn, and this is exactly what we chose to do. Based on this argument, what is $\mathbf{w}^\top \mathbf{x}$ regressing towards when we learn the weights $\mathbf{w}$ in logistic regression?

2 Multiclass Logistic Regression

Problem 1. We can try to leverage on the concept of log odds and derive a function that behaves like the logistic function but for $K > 2$. This will lead up towards deriving multiclass logistic regression using a similar probabilistic argument. However, the notion of odds only works when we have two possible outcomes. As such, let us hold out class K without loss of generality and find the odds of class $j \in \{1, \ldots, K-1\}$ against class K. Hence, we define the log odds z_j of class j against K as

$$z_j = \ln\left(\frac{p(t=j|\mathbf{x})}{p(t=K|\mathbf{x})}\right).$$

Show that $p(t=j|\mathbf{x}) = p(t=K|\mathbf{x}) \cdot \exp(z_j)$.

Problem 2. Find $p(t=K|\mathbf{x})$ in terms of only functions of $\mathbf{x}$ and $\{z_j\}_{j=1}^{K-1}$.

Problem 3. Using the results from Problem 1 and 2, show that for $j \in \{1, \ldots, K-1\}$, we have

$$p(t=j|\mathbf{x}) = \frac{\exp(z_j)}{\sum_{i=1}^{K}\exp(z_i)}.$$

What have we done so far? Recall that we are seeking for an error function for multi-class classification. This sounds like a job that MLE can accomplish. However, in order to apply MLE, we need to choose some likelihood function $p(t|\mathbf{x}, \mathbf{w})$. The narrative thus far suggest that we can use

$$p(t=j|\mathbf{x}, \mathbf{w}) = \frac{\exp(z_j)}{\sum_{i=1}^{K}\exp(z_i)}.$$

In Problem 1, we see that one approach for introducing parameters $\mathbf{w}$ into the learning process is by assuming that the log odd z_{binary} are linear in $\mathbf{x}$ (i.e. $z_{\text{binary}} = \mathbf{w}^\top\mathbf{x}$). We shall do the same here and assume that all log odds z_j are linear in $\mathbf{x}$ (i.e. $z_j = \mathbf{w}_{(j)}^\top\mathbf{x}$ for all j). Let us stack the weights into a matrix

$$\mathbf{W} = \begin{bmatrix} —\mathbf{w}_{(1)}^\top— \\ —\mathbf{w}_{(2)}^\top— \\ \vdots \\ —\mathbf{w}_{(K)}^\top— \end{bmatrix}.$$

Then you can rewrite these system of equations as $\mathbf{z} = \mathbf{W}\mathbf{x}$. With this choice of likelihood, if we are now given an input $\mathbf{x}$, we can predict the probabilities that $\mathbf{x}$ belongs to class j by simply computing $p(t=j|\mathbf{x}, \mathbf{W}) = \text{softmax}(\mathbf{W}\mathbf{x})_j$. For convenience, we shall stack these probabilities into a vector

$$\begin{bmatrix} p(t=1|\mathbf{x}, \mathbf{W}) \\ p(t=2|\mathbf{x}, \mathbf{W}) \\ \vdots \\ p(t=K|\mathbf{x}, \mathbf{W}) \end{bmatrix} = \text{softmax}(\mathbf{W}\mathbf{x}).$$

Problem 4. Now that we have a reasonable choice of the likelihood function $p(t|\mathbf{x}, \mathbf{W})$, we are now ready to perform MLE. Using the maximum likelihood paradigm, derive an error function $E(\mathbf{W})$ for the multiclass logistic regression. You may keep any `logsumexp` terms in your solution without further simplification.